MOVIDRIVE®
Drive Inverters

DeviceNet DFD11A Fieldbus Interface
Manual

Edition 11/98

10/262/97

DeviceNet

0919 6714 / 1198

TÜV CERT DIN EN ISO 9001

SEW EURODRIVE

Important Notes

- **Read this manual carefully before you start the installation and startup of MOVIDRIVE® drive inverters with the DFD11A DeviceNet option.**
  This manual assumes that the user has access to and is familiar with the documentation on the MOVIDRIVE® system, in particular the MOVIDRIVE® system manual.

- **Safety notes:**
  Always follow the safety and warning instructions contained in this manual!
  Safety notes are marked as follows:

Electrical hazard, e.g. when working on live wires.

Mechanical hazard, e.g. when working on hoists.

Important instructions for safe and fault-free operation of the driven machine/ system, e.g. pre-setting before startup.

- **General safety notes on bus systems:**
  This communication system allows you to match the MOVIDRIVE® drive inverter to the specifics of your application to a very high degree. As with all bus systems, there is a danger of invisible, external (as far as the inverter is concerned) modifications to the parameters which give rise to changes in the inverter's behavior. This may result in unexpected (not uncontrolled, though!) system behavior.

- Each unit is manufactured and tested to current SEW-EURODRIVE technical standards and specifications.
  The manufacturer reserves the right to make changes to the technical data and designs as well as the user interface herein described, which are in the interest of technical progress.
  A requirement of fault-free operation and fulfillment of any rights to claim under guarantee is that this information is observed.

Contents

1 Introduction 4

2 Installation 6
2.1 Supported unit types 6
2.2 Installation of the option card 6
2.3 Pin assignment 6
2.4 Shielding and routing of the bus cables 7
2.5 Bus termination 7
2.6 Setting the DIP switches 8
2.7 Display elements 8
2.7.1 Power-up sequence 8
2.7.2 ModNet LED 9
2.7.3 PIO LED 9
2.7.4 BIO LED 10
2.7.5 BUSOFF LED 10

3 Project Planning and Startup 11
3.1 Inverter control mode fieldbus 11
3.2 Setting up the DeviceNet network using the DeviceNet Manager software 13
3.2.1 Installing the EDS file 13
3.2.2 Connecting the device to an existing network 13
3.3 Process data exchange 14
3.3.1 Polled I/O 14
3.3.2 Bit-strobe I/O 18
3.4 Parameter data exchange 20
3.4.1 The SEW parameter data channel 20

4 Sample Application with PLC Type SLC500 26
4.1 Exchange of polled I/O (process data) 28
4.2 Exchange of bit strobe I/O 30
4.3 Exchange of explicit messages (parameter data) 31

5 Technical Data 35

6 Appendix 36
6.1 General error codes 36
6.2 Statement of conformance 37
6.3 Definitions of terminology 46

7 Index 47

1 Introduction

This user manual for the DeviceNet (DFD11A) option describes how to install the DFD11A DeviceNet option card in the drive inverter and how to start up the MOVIDRIVE® on the DeviceNet fieldbus system.

As well as explaining all settings on the fieldbus interface, this document also deals with the various DeviceNet connection variants in the form of brief startup examples.

As well as this manual on the DeviceNet option, you should request the following publications dealing with the topic of fieldbuses in more detail, so as to permit MOVIDRIVE® to be connected to the DeviceNet fieldbus system in a straightforward and effective fashion:

- "MOVIDRIVE® Fieldbus Unit Profile" manual
- "MOVIDRIVE®" system manual

The manual for the MOVIDRIVE® fieldbus unit profile describes the fieldbus parameters and their coding, as well as explaining the whole range of various control concepts and application options in the form of brief examples.

The "MOVIDRIVE®" system manual contains a listing of all parameters of the drive inverter which can be read and/or written via the various communications interfaces such as RS-485 and also via the fieldbus interface.

The MOVIDRIVE® drive inverter in conjunction with the DFD11A option enables you to make the connection to master automation systems via DeviceNet, thanks to the high-performance, universal fieldbus interface of the DFD11A option.

MOVIDRIVE® and DeviceNet

The device behavior of the inverter which forms the basis of DeviceNet operation is referred to as the device profile. It is independent of any particular fieldbus and is therefore a uniform feature. This provides you, the user, with the opportunity of developing applications irrespective of the fieldbus. As a result, it is very easy to change over to other bus systems such as INTERBUS (DFI11A option), PROFIBUS (DFP11A option) or CAN bus (DFC11A).

MOVIDRIVE® offers you direct access to all drive parameters and functions via the DeviceNet interface. The drive inverter is controlled via the high-speed, cyclical process data. Via this process data channel, you can enter setpoints such as the setpoint speed, ramp generator time for acceleration/deceleration, etc. as well as triggering various drive functions such as enable, control inhibit, normal stop, rapid stop, etc. However, at the same time you can also use this channel to read back actual values from the drive inverter, such as the actual speed, current, device status, fault number and also reference signals.

The parameters of the inverter are set exclusively by using Explicit Messages, whereas the process data exchange is replicated on the DeviceNet services of Polled I/O or Bit-Strobe I/O. This parameter data exchange enables you to implement applications in which all the important drive parameters are stored in the master programmable controller, so that there is no need for manual parameter settings on the drive inverter itself.

Every DeviceNet option card is designed so the fieldbus-specific MAC-ID and baud rate settings are made using hardware switches on the option card. This manual setting means the drive inverter can be integrated into the DeviceNet environment and switched on within a very short period of time. The parameter setting process can be performed in a completely automated fashion by the DeviceNet master (parameter download). This future-oriented variant offers the advantages of shortening the system startup time and simplifying the documentation of your application program, because all the important drive parameters can now be stored directly in your control program.

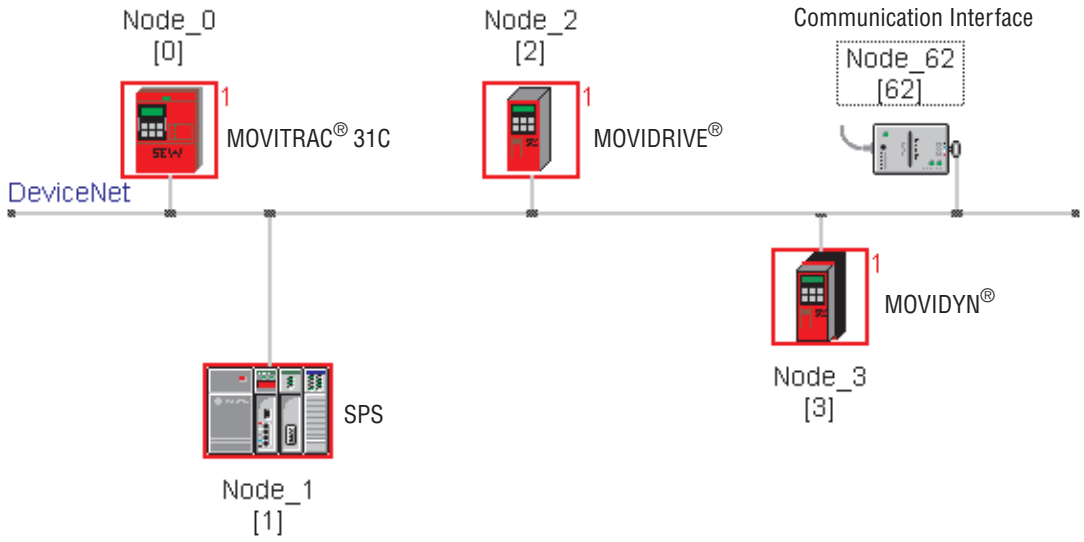


Fig. 1: DeviceNet with MOVITRAC® 31C, MOVIDRIVE®, MOVIDYN® and PLC

Using a fieldbus system requires additional monitoring functions in the drive engineering, e.g. time monitoring of the fieldbus (fieldbus timeout) as well as rapid stop concepts. For example, you can specifically adapt the monitoring functions of MOVIDRIVE® to your application. You can determine, for instance, which fault reaction of the drive inverter should be triggered in the event of a bus error. A rapid stop is a good idea for many applications, although this can also be achieved by "freezing" the last setpoints so the drive continues operating with the most recently valid setpoints (e.g. conveyor belt). The range of control terminal functions is also ensured in fieldbus mode, so you can continue to implement rapid stop concepts independent of the fieldbus by means of the drive inverter terminals.

The MOVIDRIVE® drive inverter offers you numerous diagnostic options for startup and service purposes. For example, you can use the integrated fieldbus monitor to check both the setpoints sent by the master control and the actual values.

Furthermore, you are supplied with numerous additional items of information about the status of the fieldbus option card. The fieldbus monitor function in conjunction with the MX_SHELL PC software offers you an easy-to-use diagnostic facility permitting all drive parameters to be set (including the fieldbus parameters) as well as displaying the fieldbus and device status information in detail.

2 Installation

2.1 Supported unit types

The DFD11A option for the DeviceNet connection can be operated with all drive inverters in the MOVIDRIVE®series.

2.2 Installation of the option card

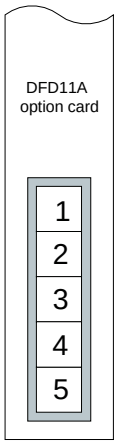
Before you begin:

- Take suitable measures to dissipate any electrical charge in your body before you touch the option card (discharge strap, conductive shoes, etc.).
- Keep the option card in its original packaging and do not remove it until it is to be installed.
- Do not touch the option card more than necessary, and only hold it by the edge of the circuit board. Do not touch any components.

Installing the option card:

- De-energize the inverter. Switch off the mains and the 24 V supply, if used.
- Remove the lower hood cover from the control module.
- Unscrew the electronics shield clamp.
- Remove the black cover plate.
- Insert the option card into the guide rails of the OPTION1 slot and push it in.
- Insert option card by applying moderate pressure on the front panel. The option card has been installed correctly when it is flush with the control card.
- Screw the electronics shield clamp back on.
- Put the hood cover of the control module back on.
- The DFD11 option card is now fully installed.

2.3 Pin assignment



02119AEN

Fig. 2: Pin assignment

The assignment of connecting terminals is described in the DeviceNet specification Volume I, Appendix A.

Pin no.	Abbrev.	Meaning	Color
1	V-	0V24	black
2	CAN_L	CAN_L	blue
3	DRAIN	DRAIN	shiny
4	CAN_H	CAN_H	white
5	V+	24V	red

Table 1: Connection terminal CAN bus

The DeviceNet option card is opto-decoupled on the driver side in accordance with the DeviceNet specification (Volume I, Chapter 9). This means the CAN bus driver must be powered with 24 V via the bus cable.

The required cable is also described in the DeviceNet specification (Volume I, Appendix B). The connection must be made according to the color code specified in Table 1.

2.4 Shielding and routing of the bus cables

Having the bus cable correctly shielded reduces electrical interference which can occur in an industrial environment. The following measures ensure the best possible shielding:

- Fasten the retaining screws of plugs, modules and potential compensating cables until finger-tight.
- Apply the bus cable shielding on both ends.
- Do not route the signal and bus cables in parallel to the power cables (motor leads); use separate cable ducts if at all possible.
- Use only metal, grounded cable racks in industrial environments.
- Route the signal cables and the associated potential compensation closely to each other at the shortest distance.
- Avoid using plug connections to extend bus cables.
- Route the bus cables in close proximity to existing grounding surfaces.
- Use bus connectors with a metal-plated or metal housing.

IMPORTANT!

In the event of fluctuations in the ground potential, a compensating current may flow along the shield which is connected at both ends and to the ground potential (PE). In this case, make adequate provision for potential compensation in accordance with the relevant VDE regulations.

2.5 Bus termination

In order to avoid disruptions in the bus system due to reflections etc., each DeviceNet segment must be terminated with 120 Ω bus terminating resistors at the first and last physical participant.

The bus terminating resistor must be wired between terminals 2 and 4 of the bus connector.

2.6 Setting the DIP switches

There are two DIP switch blocks of 4 DIP switches each on the DFD11A DeviceNet card. Two DIP switches are used for setting the baud rate and 6 for setting the MAC-ID (Media Access Control Identifier). The MAC-ID represents the node address of the DFD11A.

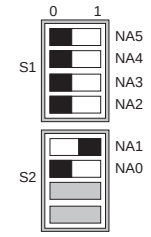
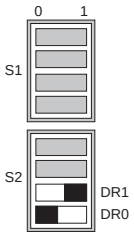
Function	Abbrev.	Number of bits	Representation	Labeling	Meaning
MAC-ID	NA	6 bits	 e.g. MAC-ID = 2	NA5..0	0..63
Baud rate	DR	2 bits	 e.g. 10: 500 kbaud	DR1..0	00: 125 kbaud 01: 250 kbaud 10: 500 kbaud 11: Invalid

Table 2: MAC-ID and baud rate setting

2.7 Display elements

The display elements comprise 4 bicolor LEDs.

Function	Abbreviation
Module/network status LED	ModNet
Polled I/O	PIO
Bit-strobe I/O	BIO
Bus off	BUSOFF

Table 3: Display elements

2.7.1 Power-up sequence

All LEDs are tested after the unit is switched on. The LEDs are switched on in the following sequence as part of the test:

Time	ModNet LED	PIO LED	BIO LED	BUSOFF LED
0 ms	Green	Off	Off	Off
250 ms	Red	Off	Off	Off
500 ms	Off	Green	Off	Off
750 ms	Off	Red	Off	Off
1000 ms	Off	Off	Green	Off
1250 ms	Off	Off	Red	Off
1500 ms	Off	Off	Off	Green
1750 ms	Off	Off	Off	Red
2000 ms	Off	Off	Off	Off

Table 4: Power-up LED test

The power-up test is always performed unless the fieldbus is powered with 24 V!

2.7.2 ModNet LED

The range of functions of the ModNet LED (module/network status LED) is defined in the DeviceNet specification. Its range of functions is described in Table 5.

Status	LED	Message
Not switched on/ off-line	Off	• Unit is in off-line status • Unit is performing DUP-MAC check • Unit is switched off
On-line and in operational mode	Flashes green (1 s cycle)	• The unit is on-line and no connection has been set up • DUP-MAC check was performed successfully • No connection has yet been established with a master • No configuration, wrong configuration or configuration not complete
On-line, operational mode and connected	Green	• On-line • Connection has been established with a master • Connection is active (established state)
Minor fault or connection timeout	Flashes red (1 s cycle)	• A correctable error has occurred • Polled I/O or/and bit-strobe I/O connections are in timeout status • A correctable error has occurred in the unit
Critical fault or critical link failure	Red	• A non-correctable error has occurred • BusOff • DUP-MAC check has detected an error

Table 5: Status table of the ModNet LED

2.7.3 PIO LED

The PIO LED checks the polled I/O connection (process data channel). Its range of functions is described in Table 6.

Status	LED	Message
DUP-MAC check	Flashes green (125 ms cycle)	• Unit is performing the DUP-MAC check
Not switched on/ off-line but not DUP-MAC check	Off	• Unit is in off-line status • Unit is switched off
On-line and in operational mode	Flashes green (1 s cycle)	• The unit is on-line • DUP-MAC check was performed successfully • A PIO connection is being established with a master (configuring status) • No configuration, wrong configuration or configuration not complete
On-line, operational mode and connected	Green	• On-line • A PIO connection has been established (established status)
Minor fault or connection timeout	Flashes red (1 s cycle)	• A correctable error has occurred • Polled I/O connection is in timeout status
Critical fault or critical link failure	Red	• A non-correctable error has occurred • BusOff • DUP-MAC check has detected an error

Table 6: Status table of the PIO LED

2.7.4 BIO LED

The BIO LED checks the bit-strobe I/O connection. Its range of functions is described in Table 7.

Status	LED	Message
DUP-MAC check	Flashes green (125 ms cycle)	• Unit is performing the DUP-MAC check
Not switched on/ off-line but not DUP-MAC check	Off	• Unit is in off-line status • Unit is switched off
On-line and in operational mode	Flashes green (1 s cycle)	• The unit is on-line • DUP-MAC check was performed successfully • A BIO connection is being established with a master (configuring state) • No configuration, wrong configuration or configuration not complete
On-line, operational mode and connected	Green	• On-line • A BIO connection has been established (established state)
Minor fault or connection timeout	Flashes red (1 s cycle)	• A correctable error has occurred • Bit-strobe I/O connection is in timeout state
Critical fault or critical link failure	Red	• A non-correctable error has occurred • BusOff • DUP-MAC check has detected an error

Table 7: Status table of the BIO LED

2.7.5 BUSOFF LED

The BUSOFF LED displays the physical status of the bus node. Its range of functions is described in Table 8.

Status	LED	Message
NO ERROR	Off	• The number of bus errors is in the normal range (error active state).
BUS WARNING	Flashes red (125 ms cycle)	• The unit is performing a DUP-MAC check and cannot send any messages because no other participants are connected to the bus (error passive state).
	Flashes red (1 s cycle)	• The number of physical bus errors is too high. No more error telegrams are actively written to the bus (error passive state).
BUS ERROR	Red	• BusOff status • The number of physical bus errors has continued to grow despite the switch to the error passive state. Access to the bus is deactivated.

Table 8: Status table of the BUSOFF LED

3 Project Planning and Startup

This chapter describes how to start up the MOVIDRIVE® inverter with the DFD11A option, using MX_SHELL (from version 1.30) or the DBG11A keypad (from version .13).

3.1 Inverter control mode fieldbus

After the DeviceNet option card has been installed and the baud rate and MAC-ID have been set (using the DIP switches), the parameters for the MOVIDRIVE® inverter can be set immediately via the fieldbus system without further manual adjustment. All drive parameters can be downloaded in this manner from the master programmable controller directly via DeviceNet after the power is switched on.

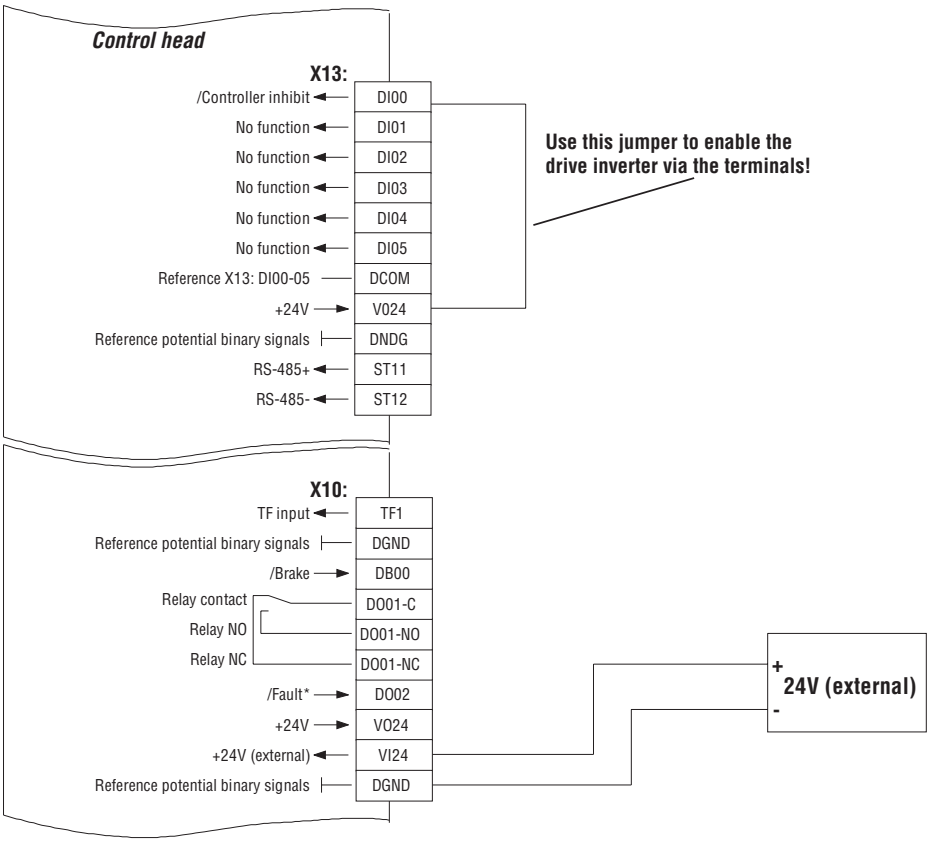
The inverter has to be switched to the appropriate *control mode* prior to control via DeviceNet. This can be done with parameter P100 *Setpoint source* and P101 *Control signal source*. The factory settings for these parameters are the values *Bipol./Fix. setp.* or *Terminals* (setpoint processing via analog setpoint and control via input terminals). The setting P100 Setpoint source = *FIELDBUS* causes the inverter to get its setpoints from the fieldbus; if P101 = *FIELDBUS*, the inverter is controlled via the fieldbus (i.e. enable, rapid stop, normal stop, controller inhibit, etc.). MOVIDRIVE® will now respond to the process output data transmitted from the master programmable controller.

Activation of the *fieldbus* control mode is signalled to the master control by the *PA data enabled* bit in the status word.

For safety reasons, the inverter must also be enabled on the terminal side for control via the fieldbus system. Accordingly, the terminals must be wired up or programmed in such a way that the inverter is enabled via the input terminals. The easiest way of enabling the inverter on the terminal side is to wire input terminal DI00 (/CONTROLLER INHIBIT function) to the +24 V signal and program input terminals DI01 to DI05 to NO FUNCTION.

Procedure for startup of the MOVIDRIVE® inverter with fieldbus interface:

1. **Switch the drive inverter to ENABLE on the terminal side**
   Wire input terminal DI00 (/CONTROLLER INHIBIT function) to the +24 V signal (e.g. with a jumper).



01267AEN

2. **Control mode = FIELDBUS**
   Use parameters P100 and P101 to switch control and setpoint processing of the servo inverter to FIELDBUS.

P100	**Setpoint source**	**FIELDBUS**
P101	**Control signal source**	**FIELDBUS**

3. **Input terminal DI01 = NO FUNCTION**
   Input terminal DI02 = NO FUNCTION
   Input terminal DI03 = NO FUNCTION
   Input terminal DI04 = NO FUNCTION
   Input terminal DI05 = NO FUNCTION
   Program the function of input terminals DI01 to DI05 to NO FUNCTION with parameters P600 to P604.

P600	**Binary input DI01**	**NO FUNCTION**
P601	**Binary input DI02**	**NO FUNCTION**
P602	**Binary input DI03**	**NO FUNCTION**
P603	**Binary input DI04**	**NO FUNCTION**
P604	**Binary input DI05**	**NO FUNCTION**

3.2 Setting up the DeviceNet network using the DeviceNet Manager software

3.2.1 Installing the EDS file

Fig. 3: EDS file list

There is an EDS file **MDX.eds** and a bitmap file **MDX.bmp** for the DFD11A option card. These files must be installed using the DeviceNet Manager software.

To do this, select *Utilities/Install EDS File* from the menu. The program then prompts you for the EDS file name and the bitmap file. The EDS file is then installed. More details on the installation of the EDS file can be found in the Allen Bradley documentation for the DeviceNet Manager.

After installation, the device is available in the device list under the entry *SEW Geräteprofil/SEW-Eurodrive GmbH/SEW-MOVIDRIVE-DFD11A*.

You can use the following Internet addresses for obtaining current EDS files and for further information about DeviceNet.

- SEW-EURODRIVE: www.sew-eurodrive.de
- Allen Bradley: www.ab.com
- Rockwell: www.rockwell.com
- Open Device Net Vendor Association: www.odva.org

3.2.2 Connecting the device to an existing network

All EDS files are automatically read in after the DeviceNet Manager software is called up. The device list contains all devices which have been defined by an EDS file.

3.3 Process data exchange

3.3.1 Polled I/O

The polled I/O messages correspond to the process data messages of the SEW fieldbus profile and up to three process data words can be exchanged between the control and the inverter.

The process data length can be set with MX_SHELL (version 1.30 or later) and DBG (version .13 or later) using the parameter *P877 DeviceNet PD Configuration*. The inverter must be switched off and on again after this parameter has been changed, in order to activate the set process data length.

The process data length can also be set using the parameter data channel of DeviceNet. The process data configuration takes effect immediately, if the process data length is set using the parameter data channel.

IMPORTANT:

The set process data length determines the process data lengths of both the polled I/O and the bit-strobe I/O messages, i.e. the settings for the process data lengths of both the polled I/O and the bit-strobe I/O always have to be identical in the control.

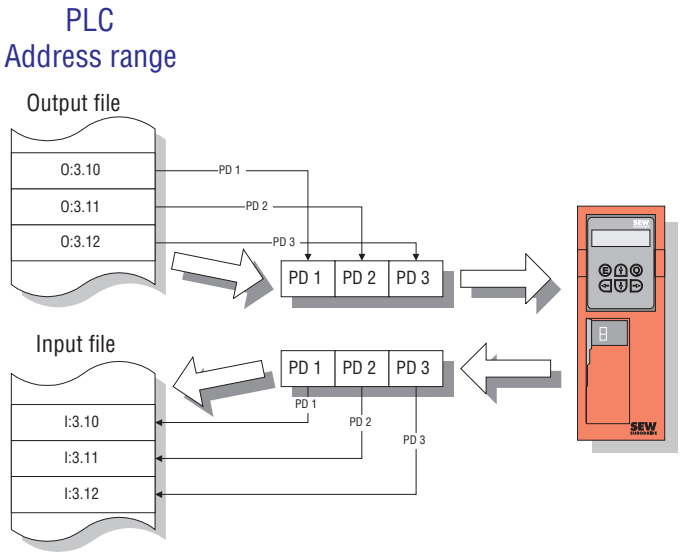
Project planning for three process data words

The factory setting of the process data configuration is process data length = 3. This setting be altered using the *DeviceNet PD Configuration* parameter.

In MX_SHELL or DBG, the setting is displayed via the parameter *DeviceNet PD configuration = 3 PD* or *3PD + Param*. As a result, three process data words (6 bytes) are processed in the inverter and three process input data words are sent to the control.

No process data are processed or sent back, if the control transmits more than three process output data words.

Three process data words are processed, and three process input data words are sent from the inverter to the control, if the control transmits three process output data words.



02120AEN

Fig. 4: Representation of three process data words in the PLC memory area

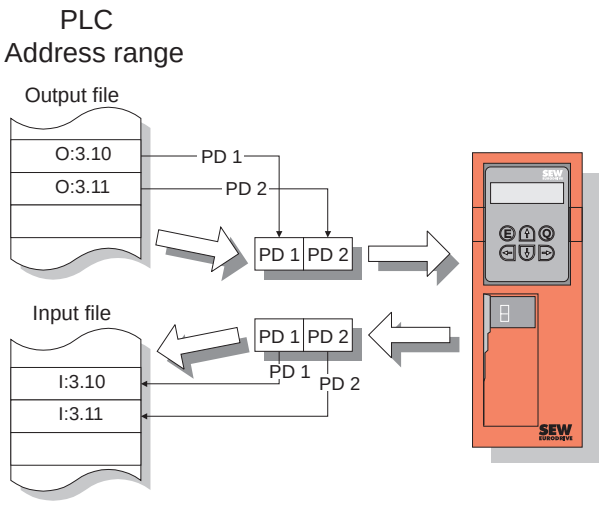
The process output data are stored in the PLC output file and the process input data of the PLC are stored in the input file. In the example above, the output data words O:3.10, O:3.11 and O:3.12 are copied to process output data words 1, 2 and 3 and processed by the inverter. The inverter sends back three process input data words which are copied into input data words I:3.10, I:3.11 and I:3.12 of the PLC.

Project planning for two process data words

Process data length = 2 can be set using the *DeviceNet PD Configuration* parameter. In this case, the setting *2 PD* or *2PD + Param* must be selected in MX_SHELL or DBG. As a result, two process data words (4 bytes) are processed in the inverter and two process input data words are sent to the control.

No process data are processed or sent back, if the control transmits more than two process output data words.

Two process data words are processed, and two process input data words are sent from the inverter to the control, if the control transmits two process output data words.



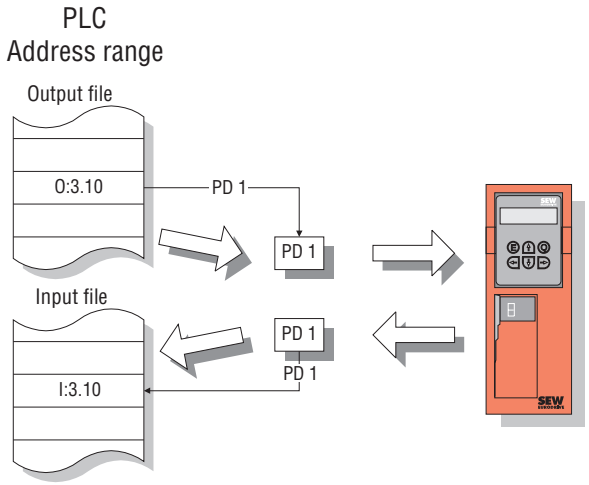
02121AEN

Fig. 5: Representation of two process data words in the PLC memory area

The process output data are stored in the PLC output file and the process input data of the PLC are stored in the input file. In the example above, the output data words O:3.10 and O:3.11 are copied to process output data words 1 and 2 and processed by the inverter. The inverter sends back two process input data words which are copied into input data words I:3.10 and I:3.11 of the PLC.

Project planning for one process data word

Process data length = 1 can be set using the *DeviceNet PD Configuration* parameter. In this case, the setting *1 PD* or *1PD + Param* must be selected in MX_SHELL or DBG. As a result, one process output data word (1 byte) is processed in the inverter and one process input data word is sent to the control. The control is permitted to send only one process output data word. No process data are processed or sent back if more than one process output data word is sent by the control.



02122AEN

Fig. 6: Representation of one process data word in the PLC memory area

The process output data are stored in the PLC output file and the process input data of the PLC are stored in the input file. In the example above, output data word O:3.10 is copied to process output data word 1 and processed by the inverter. The inverter sends back one process input data word which is copied to input data word I:3.10 of the PLC.

Timeout response with polled I/O

The timeout response is triggered by the DeviceNet option card. The timeout interval must be set by the master after the connection has been established. The DeviceNet specification refers to an “expected packet rate” rather than a timeout interval. The expected packet rate is calculated on the basis of the timeout interval using the following formula:

$$\text{Timeout_Inverter} = t_{\text{Timeout_Interval_PolledIO}} = 4 \cdot t_{\text{Expected_Packet_Rate_PolledIO}}$$

The expected packet rate can be set using the connection object class 5, instance 2, attribute 9. The range of values runs from 0 ms to 65535 ms in 5 ms steps.

The expected packet rate for the polled I/O connection is converted into the timeout interval and displayed in the device and the timeout interval in parameter *P819*.

This timeout interval is retained in the device whenever the polled I/O connection is dropped, and the device switches to timeout status after the timeout interval has elapsed.

The timeout interval must not be altered in the inverter using MX_SHELL or the DBG, because it can only be activated via the bus.

If a timeout occurs for the polled I/O messages, this connection type enters timeout status. Incoming polled I/O messages are no longer accepted.

The timeout response triggers timeout reaction set in the inverter.

The timeout response can be reset with DeviceNet by using the reset service of the connection object (class 0x05, instance 0x02, undetermined attribute), by dropping the connection, by using the reset service of the identity object (class 0x01, instance 0x01, undetermined attribute) or with the reset bit in the control word.

3.3.2 Bit-strobe I/O

Bit-strobe I/O messages are not contained in the SEW fieldbus profile. They represent a process data exchange which is specific to DeviceNet.

The master sends out a broadcast message that is 8 bytes = 64 bits long. One bit in this message is assigned to each participant in accordance with its station address. The value of this bit may be 0 or 1, triggering two different reactions in the recipient.

Bit value	Message	BIO LED
0	Only send back the process input data	Continuously green
1	Trigger fieldbus timeout response and send back process input data	Continuously green

Table 9: Bit strobe signal messages

IMPORTANT:

The BIO LED can be consulted to distinguish between the timeout triggered by the bit-strobe message and a real timeout in the connection. It remains continuously green if the timeout is triggered by the bit-strobe message.

If the BIO LED flashes red, there is a timeout in the bit-strobe connection and no additional bit-strobe messages are accepted.

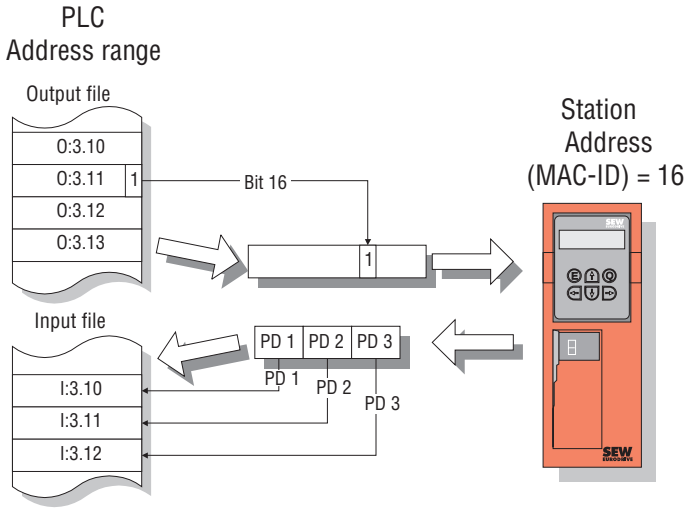
Table 10 shows the data area of the bit-strobe request message which represents the allocation of participants (= station address) to data bits.

For example, the participant with station address (MAC-ID) 16 only processes bit 0 in data byte 2.

Byte offset	7	6	5	4	3	2	1	0
0	ID 7	ID 6	ID 5	ID 4	ID 3	ID 2	ID 1	ID 0
1	ID 15	ID 14	ID 13	ID 12	ID 11	ID 10	ID 9	ID 8
2	ID 23	ID 22	ID 21	ID 20	ID 19	ID 18	ID 17	ID 16
3	ID 31	ID 30	ID 29	ID 28	ID 27	ID 26	ID 25	ID 24
4	ID 39	ID 38	ID 37	ID 36	ID 35	ID 34	ID 33	ID 32
5	ID 47	ID 46	ID 45	ID 44	ID 43	ID 42	ID 41	ID 40
6	ID 55	ID 54	ID 53	ID 52	ID 51	ID 50	ID 49	ID 48
7	ID 63	ID 62	ID 61	ID 60	ID 59	ID 58	ID 57	ID 56

Table 10: MAC-ID assignment in the bit-strobe I/O message request

Each participant which has received this bit-strobe I/O message responds with its current process input data. The length of the process input data corresponds to the process data length for the polled I/O connection. It can also be set using the *DeviceNet PD Configuration* parameter.



02123AEN

Fig. 7: Bit-strobe I/O messages

In the example above, the bit-strobe I/O message is stored in memory words O:3.10 to O:3.13 and Bit 16 is assigned to the device with station address 16. This bit corresponds to bit 0 of output data word O:3.11 in the PLC. Since this bit was set to the value 1, a fieldbus timeout is triggered in the inverter with station address 16. However, the BIO LED remains continuously green to indicate that the bit-strobe connection is not in timeout status.

The inverter sends three process input data words to the PLC; these are stored in input data words I:3.10 to I:3.12 in the input file.

IMPORTANT:

The set process data length determines the process data length of both the bit-strobe I/O and the polled I/O messages, i.e. the settings for the process data length of both the polled I/O and the bit-strobe I/O always have to be identical in the control.

Timeout response with bit-strobe I/O

The timeout response is triggered by the DeviceNet option card. The timeout interval must be set by the master after the connection has been established. The DeviceNet specification refers to an "expected packet rate" rather than a timeout interval. The expected packet rate is calculated on the basis of the timeout interval using the following formula:

$$t_{Timeout_Interval_BitStrobeIO} = 4 \cdot t_{Expected_Packet_Rate_BitStrobeIO}$$

It can be set using connection object class 5, instance 3, attribute 9. The range of values runs from 0 ms to 65535 ms in 5 ms steps.

If a timeout occurs for the bit-strobe I/O messages, this connection type enters timeout status. Incoming bit-strobe I/O messages are no longer accepted.

The timeout response is not transmitted to the inverter.

The timeout response can be reset with DeviceNet using the reset service of the connection object (class 0x05, instance 0x03, undetermined attribute), by dropping the connection or by using the reset service of the identity object (class 0x01, instance 0x01, undetermined attribute).

3.4 Parameter data exchange

3.4.1 The SEW parameter data channel

The SEW parameter data channel represents a connection via which parameters in the inverter can be changed or read. This channel is represented by explicit messages on the DFD11A DeviceNet option card.

Access to the SEW parameter data channel is by means of the register object (class 7) and the parameter object (class 15).

Register object class (class 7)

The SEW parameter data channel can be addressed using the services *Get_Attribute_Single* and *Set_Attribute_Single*. The following possibilities for addressing the parameter data channel derive from the way the register object is specified by DeviceNet so INPUT objects can only be read and OUTPUT objects can be read and written.

Instance	INPUT/OUTPUT	Resulting MOVILINK service with	
		Get_Attribut_Single	Set_Attribut_Single
1	INPUT	READ	Invalid
2	OUTPUT	READ	WRITE
3	OUTPUT	READ	WRITE VOLATILE
4	INPUT	READ MINIMUM	Invalid
5	INPUT	READ MAXIMUM	Invalid
6	INPUT	READ DEFAULT	Invalid
7	INPUT	READ SCALING	Invalid
8	INPUT	READ ATTRIBUTE	Invalid

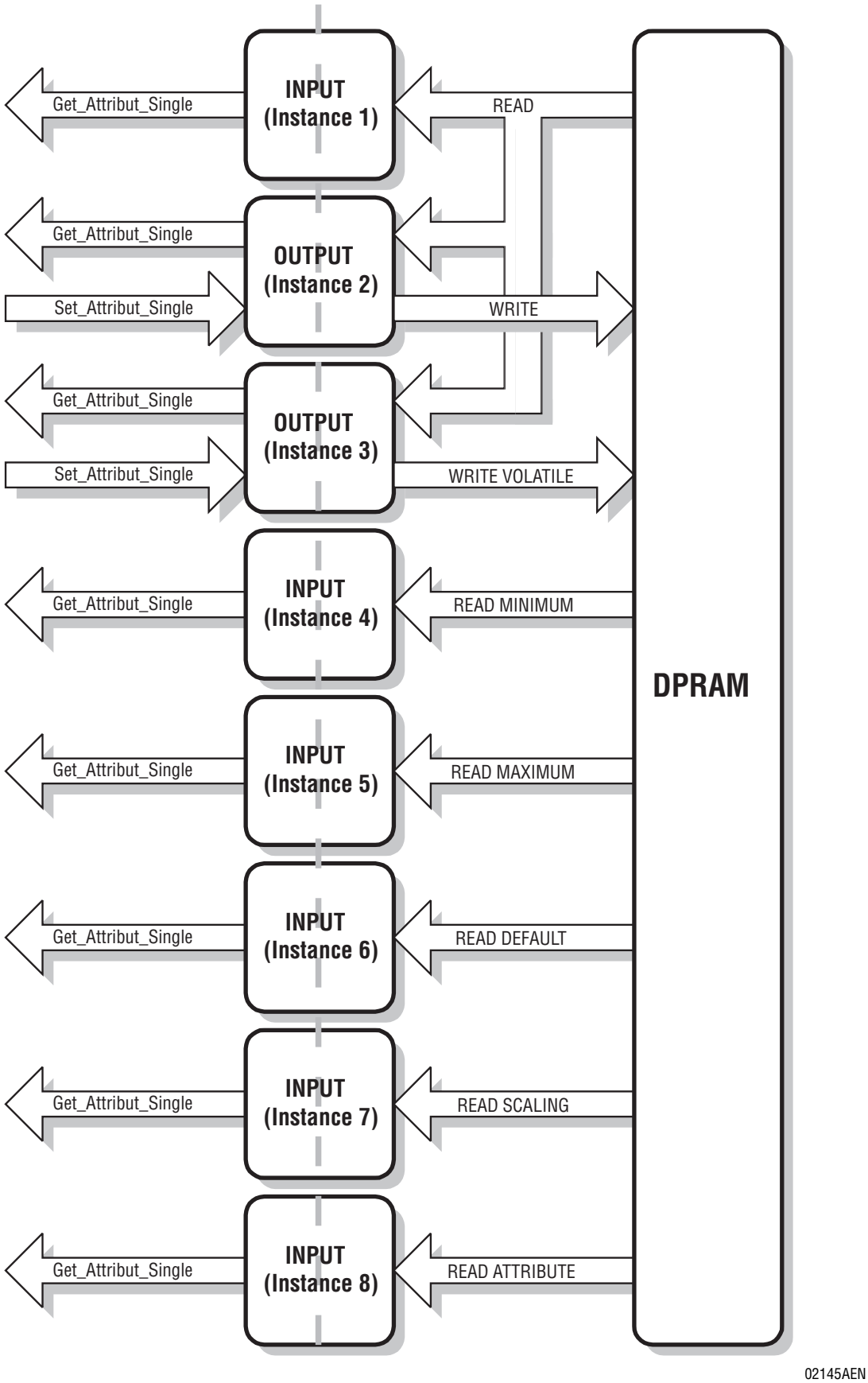


Fig. 8: Description of the parameter channel

The following table shows the attributes of the eight register object instances.

Class	Instance	Attribute	Get	Set	Type	Type/value	Meaning
0x07	0x01 (Read)	1	X		BOOL	0/1	Bad flag
		2	X		BOOL	0 (Input)	Direction
		3	X		UINT	16 bit	Size
		4	X	X	ARRAY BITS	2 byte index 4 byte data	Data
	0x02 (Read/ Write)	1	X		BOOL	0/1	Bad flag
		2	X		BOOL	1 (Output)	Direction
		3	X		UINT	48 bit	Size
		4	X	X	ARRAY BITS	2 byte index 4 byte data	Data
	0x03 (Read/ Write-Volatile)	1	X		BOOL	0/1	Bad flag
		2	X		BOOL	1 (Output)	Direction
		3	X		UINT	48 bit	Size
		4	X	X	ARRAY BITS	2 byte index 4 byte data	Data
	0x04 (Read Mini-mum)	1	X		BOOL	0/1	Bad flag
		2	X		BOOL	0 (Input)	Direction
		3	X		UINT	16 bit	Size
		4	X	X	ARRAY BITS	2 byte index 4 byte data	Data
	0x05 (Read Maxi-mum)	1	X		BOOL	0/1	Bad flag
		2	X		BOOL	0 (Input)	Direction
		3	X		UINT	16 bit	Size
		4	X	X	ARRAY BITS	2 byte index 4 byte data	Data
	0x06 (Read Default)	1	X		BOOL	0/1	Bad flag
		2	X		BOOL	0 (Input)	Direction
		3	X		UINT	16 bit	Size
		4	X	X	ARRAY BITS	2 byte index 4 byte data	Data
	0x07 (Read Scaling)	1	X		BOOL	0/1	Bad flag
		2	X		BOOL	0 (Input)	Direction
		3	X		UINT	16 bit	Size
		4	X	X	ARRAY BITS	2 byte index 4 byte data	Data
	0x08 (Read Attribute)	1	X		BOOL	0/1	BBad flag
		2	X		BOOL	0 (Input)	Direction
		3	X		UINT	16 bit	Size
		4	X	X	ARRAY BITS	2 byte index 4 byte data	Data

Table 11: Register object class

Attribute 1 *Bad flag* signals whether an error occurred in the previous service.

Attribute 2 represents the direction of the instance and attribute 3 states the data length in bits.

The actual parameter data are represented in attribute 4. They are made up of the index (2 bytes) and the data (4 bytes).

The services *Get_Attribute_Single* and *Set_Attribute_Single* are available for transmission.

Service	Coding	Comment
Get_Attribut_Single	0x0E	Read attribute
Set_Attribut_Single	0x10	Write attribute

Table 12: Register class services

Example:

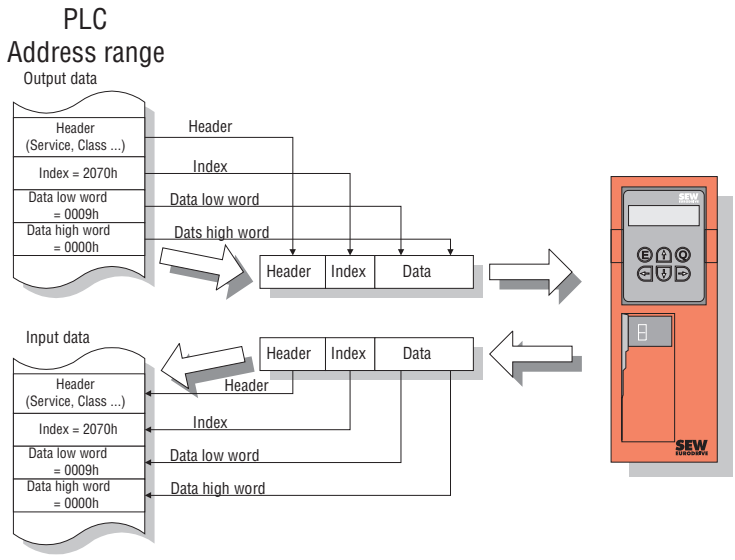
Parameter *Setpoint description PO1* (index 8304 = 2070h) is to be written with the value *CONTROL WORD1 (9)*.

Byte offset	0	1	2	3	4	5	6	7	8	9	10
Function	MAC-ID	Service	Class	Instance	Attribute	Index		Data			
Significant.						Low	High	LSB			MSB
Example	01h	10h	07h	02h	04h	70h	20h	09h	00h	00h	00h

Table 13: Data format for the parameter request message

Byte offset	0	1	2	3	4	5	6	7
Function	MAC-ID	Service	Index		Data			
Significant.			Low	High	LSB			MSB
Example	01h	90h	70h	20h	09h	00h	00h	00h

Table 14: Data format for the parameter response message



02124AEN

Fig. 9: Parametric data exchange

Parameter object class (class 15)

The fieldbus parameters of the SEW inverter can be addressed directly via the instance with the parameter object. In order to comply with the DeviceNet specification, the data format for these instances deviates from the SEW fieldbus profile.

However, it is also possible to address all parameters of the inverter via the parameter object. Instances 1 to 8 are reserved for this.

General SEW parameter data channel

Parameters are written/read via the parameter object in two steps.
1st step: Writing the index via instance 1
2nd step: Writing/reading the data via instances 2 to 8

No.	Group	Name	Comment
1	SEW parameter channel	SEW-Param.-Index	Index of the parameter
2	SEW parameter channel	SEW-Read/Write	Read or write the value of the parameter
3	SEW parameter channel	SEW-Read/WriteVo	Read or write the value of the parameter, but do not store permanently
4R	SEW parameter channel	SEW-Minimum	Read the minimum of the parameter
5R	SEW parameter channel	SEW-Maximum	Read the maximum of the parameter
6R	SEW parameter channel	SEW-Default	Read the default value of the parameter
7R	SEW parameter channel	SEW-Scaling	Read the scaling of the parameter
8R	SEW parameter channel	SEW-Attribute	Read the attributes of the parameter

Table 15: SEW parameter data channel

Consequently, two services have to be performed in order to read/write a parameter.
The data format of the *SEW param. data* corresponds to MOVILINK® (see list of parameters).

Writing/reading the fieldbus parameters

The parameters required for operating the fieldbus have been directly incorporated into the parameter object. They can be addressed directly via the instance.

No.	Group	Name	Comment
9	Communication	Control source	Control signal source
10	Communication	Setpoint source	Setpoint source
11R	Communication	PD configuration	Process data configuration
12	Communication	Setp.descr.PO1	Process output data assignment for PD1
13	Communication	Setp.descr.PO2	Process output data assignment for PD2
14	Communication	Setp.descr.PO3	Process output data assignment for PD3
15	Communication	Act.v.descr. PI1	Process input data assignment for PD1
16	Communication	Act.v.descr. PI2	Process input data assignment for PD2
17	Communication	Act.v.descr. PI3	Process input data assignment for PD3
18	Communication	PO data enable	Enable process data
19	Communication	Timeout response	Timeout response
20R	Communication	Fieldbus type	Fieldbus type
21R	Communication	Baud rate	Baud rate via DIP switch
22R	Communication	Station address	MAC-ID via DIP switch
23R	Monitor	PO1 setpoint	Monitor of process output data word 1
24R	Monitor	PO2 setpoint	Monitor of process output data word 2
25R	Monitor	PO3 setpoint	Monitor of process output data word 3
26R	Monitor	PI1 actual value	Monitor of process input data word 1
27R	Monitor	PI2 actual value	Monitor of process input data word 2
28R	Monitor	PI3 actual value	Monitor of process input data word 3

R = Read Only

Table 16: Fieldbus parameters

The scaling and the representation within the message corresponds to the DeviceNet specification (see the DeviceNet specification for coding).

Return codes for parameter setting

SEW-specific return codes

The return codes which the inverter sends back in the event of faulty parameter setting are described in the manual for the fieldbus device profile and therefore do not form part of this documentation. However, the return codes are sent back in a different format in conjunction with DeviceNet.

Byte offset	0	1	2	3
Function	MAC-ID	Service code [= 94h]	General error Code	Addi-tional Code

Table 17: Data format for the parameter response message

The *service code* of an error message is always *94h (hex).*

The *general error code* of an inverter-specific return code is always *1Fh = proprietary error.*

The additional code is identical to the *additional code low* described in the *manual for the fieldbus device profile.*

The example shows the proprietary error *10h = Impermissible parameter index*.

Return codes from DeviceNet

DeviceNet-specific return codes are sent in the error message, if the data format is not maintained during transmission or if a service is performed which has not been implemented. The coding of these return codes is described in the DeviceNet specification (see appendix).

Timeout of explicit messages

The timeout response is triggered by the DeviceNet option card. The timeout interval must be set by the master after the connection has been established. The DeviceNet specification refers to an "expected packet rate" rather than a timeout interval. The expected packet rate is calculated on the basis of the timeout interval using the following formula:

$$t_{Timeout_Interval_ExplicitMessages} = 4 \cdot t_{Expected_Packet_Rate_ExplicitMessag}$$

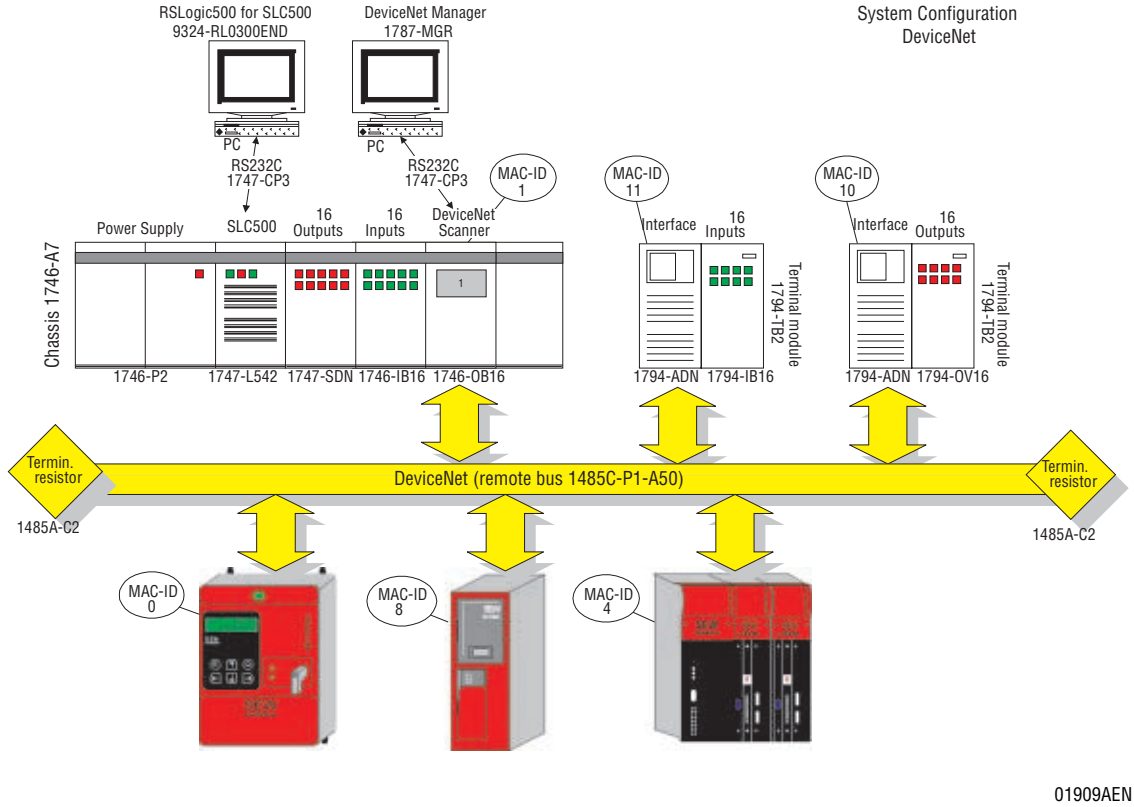
It can be set using connection object class (5), instance 1, attribute 9. The range of values runs from 0 ms to 65535 ms in 5 ms steps.

If a timeout response is triggered for the explicit messages, the connection for the explicit messages is automatically dropped providing the polled I/O or bit-strobe I/O connections are not in the ESTABLISHED status. This is the default setting for DeviceNet. The connection for explicit messages must be re-established in order to be able to communicate with these messages again.

The timeout response is <u>not</u> transmitted to the inverter.

4 Sample Application with PLC Type SLC500

The sample applications are based on a system configuration as shown in Fig. 10.



01909AEN

Fig. 10: PLC system configuration

The following devices are employed:

Device	MAC-ID
SLC5/04	---
DeviceNet scanner 1747-SDN	1
INPUT module with 32 inputs	---
OUTPUT module with 32 outputs	---
DeviceNet adapter with input module with 16 inputs	11
DeviceNet adapter with output module with 16 outputs	10
MOVITRAC with FFD31C (optional)	0
MOVIDYN with AFD11A (optional)	4
MOVIDRIVE with DFD11A	8

Table 18: Devices used in the sample application

The following memory areas have been specified with the help of the DeviceNet manager software:

```
*************************************************************************
1747-SDN Scanlist Map
*************************************************************************

Discrete Input Map:

         15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00
        ------ ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---
I:3.000  R  R  R  R  R  R  R  R  R  R  R  R  R  R  R  R  Status word of the scanner
I:3.001 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11  Process data from device 11
I:3.002 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11  Process data from device 11
I:3.003 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10  Process data from device 10
I:3.004 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10  Process data from device 10
I:3.005 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08  PED1 device 8 polled IO
I:3.006 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08  PED2 device 8 polled IO
I:3.007 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08  PED3 device 8 polled IO
I:3.008 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08  PED1 device 8 bit-strobe IO
I:3.009 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08  PED2 device 8 bit-strobe IO
I:3.010 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08  PED3 device 8 bit-strobe IO
I:3.011 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  PED1 device 0 polled IO
I:3.012 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  PED2 device 0 polled IO
I:3.013 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  PED3 device 0 polled IO
I:3.014 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  PED1 device 0 bit-strobe IO
I:3.015 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  PED2 device 0 bit-strobe IO
I:3.016 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  PED3 device 0 bit-strobe IO
I:3.017 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04  PED1 device 4 polled IO
I:3.018 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04  PED2 device 4 polled IO
I:3.019 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04  PED3 device 4 polled IO
I:3.020 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04  PED1 device 4 bit-strobe IO
I:3.021 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04  PED2 device 4 bit-strobe IO
I:3.022 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04  PED3 device 4 bit-strobe IO


Discrete Output Map:

        15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00
        ------ ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---
O:3.000  R  R  R  R  R  R  R  R  R  R  R  R  R  R  R  R  Control word of the scanner
O:3.001 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11  Process data to device 11
O:3.002 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10  Process data to device 10
O:3.003 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08  PAD1 device 8 Polled IO
O:3.004 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08  PAD2 device 8 Polled IO
O:3.005 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08  PAD3 device 8 Polled IO
O:3.006 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  PAD1 device 0 polled IO
O:3.007 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  PAD2 device 0 polled IO
O:3.008 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  PAD3 device 0 polled IO
O:3.009 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04  PAD1 device 4 polled IO
O:3.010 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04  PAD2 device 4 polled IO
O:3.011 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04  PAD3 device 4 polled IO
O:3.012 .. .. .. .. .. .. .. .. .. .. .. .. .. .. .. 08  Bit-strobe for device 8
```

The polled I/O data are displayed in *italics* and the bit strobe data in ***bold italics***.

4.1 Exchange of polled I/O (process data)

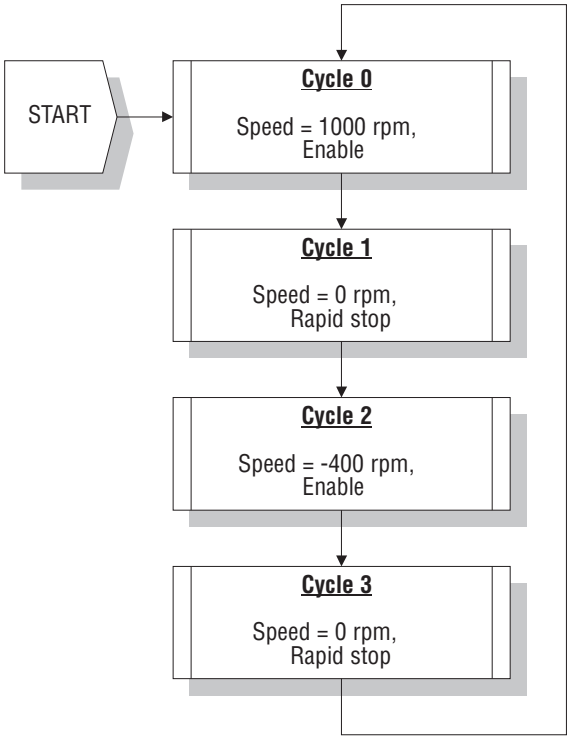


Fig. 11: Program sequence

Objective

In the following program, process data are to be sent to MOVIDRIVE® and the motor should run at a different speed.

The program sequence is shown in Fig. 11.

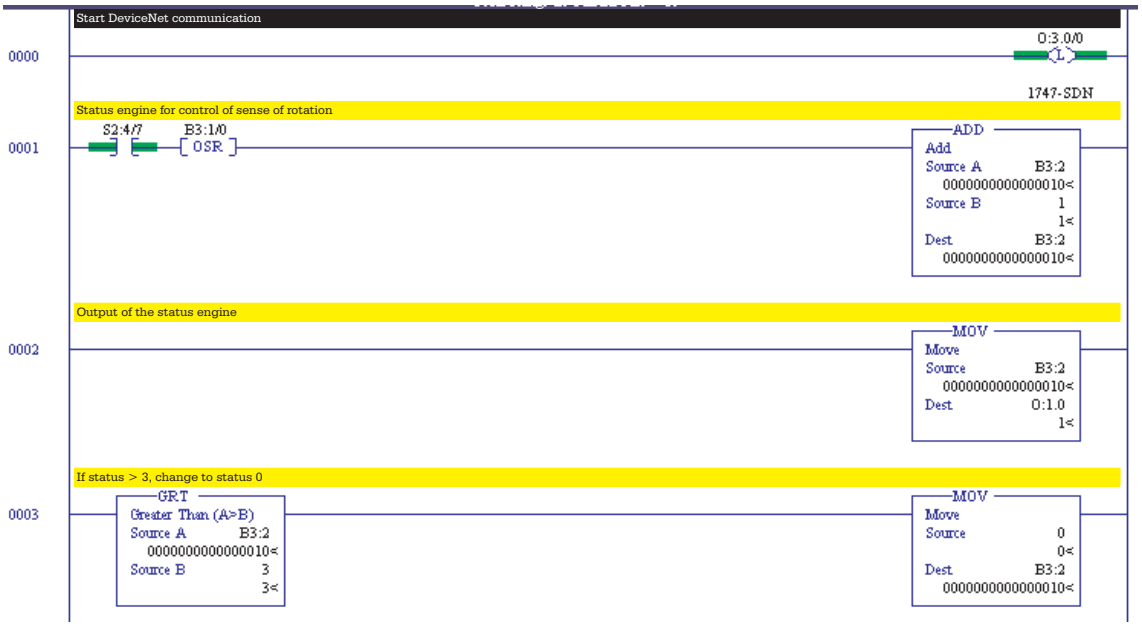
The program requires that the parameters listed in Table 19 must be set in the drive inverter.

Menu no.	Index	Parameters	Value
100	8461	Setpoint source	Fieldbus
101	8462	Control signal source	Fieldbus
870	8304	Process output data description 1	Control word 1
871	8305	Process output data description 2	Speed
872	8306	Process output data description 3	No function
873	8307	Process input data description 1	Status word 1
874	8308	Process input data description 2	Speed
875	8309	Process input data description 3	No function
876	8622	PO data enable	YES

Table 19: Setting the parameters for process data exchange

The MOVIDRIVE®drive inverter is now in fieldbus mode and can receive process data.

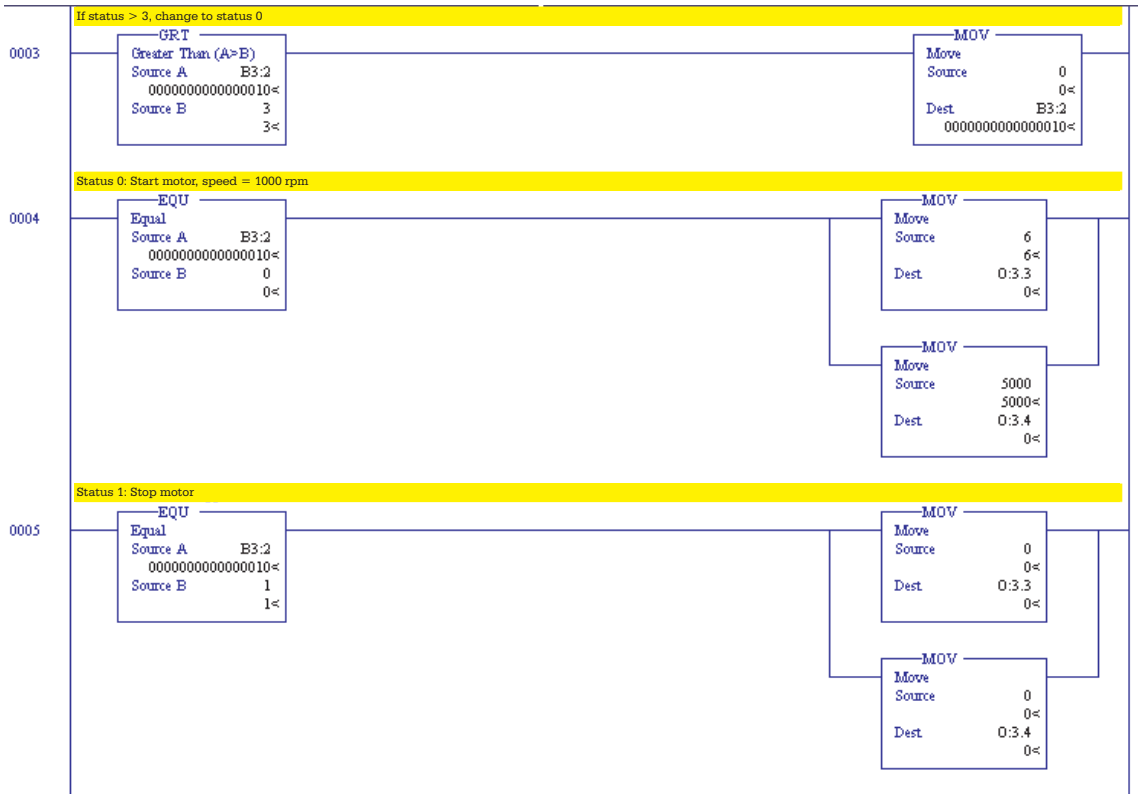
The program can now be written for the SLC500.



01912AEN

Output bit O:3.0/0 is set in rung 0 (program line 0), thereby starting DeviceNet communication (see the description of the DeviceNet scanner).

Rungs 1 and 3 implement the status engine with which states 0 – 3 are implemented. The current status is written to the outputs O:1.0 of the SLC500 output module in rung 2.

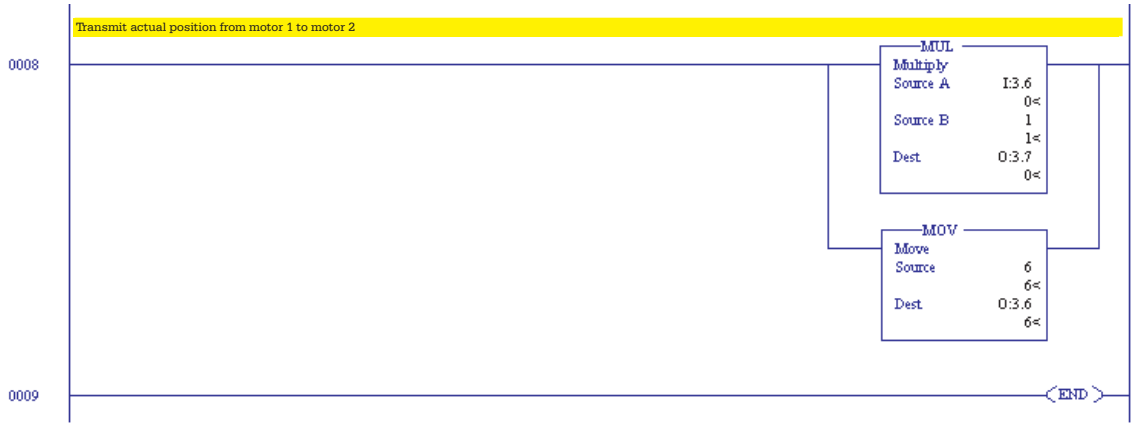


01913AEN

The figure illustrates the output of process data values to the scanner memory area.

Status 0 is created in rung 4. In this status, a 6 (ENABLE) is written to memory area O:3.3 which represents process output data word 1. A 5000 is written to memory area O:3.4 (process output data word 2), which represents 1000 rpm. This means the motor runs at 1000 rpm.

Status 1 is created in rung 5. In this status, a 0 (RAPID STOP) is written to memory area O:3.3 which represents process output data word 1. A 0 is written to memory area O:3.4 (process output data word 2), which represents 0 rpm. This means the motor is stopped with the rapid stop. States 2 and 3 are treated similarly to states 0 and 1, and are thus not explained any further.



01914AEN

In the figure above, the current actual value of the device with address 8, which is located in memory area I:3.6 (process input data word 2), is multiplied by a constant factor (in this case, by 1) and written to output memory area O:3.7 (process output data word 2 of the device with address 0).

In addition, the value 6 (ENABLE) is written to the process output data word 1 of the device with address 0 (O:3.6). Thus, the device with address 0 follows the actual speed with enable signal from the device with address 8.

4.2 Exchange of bit strobe I/O

Objective:

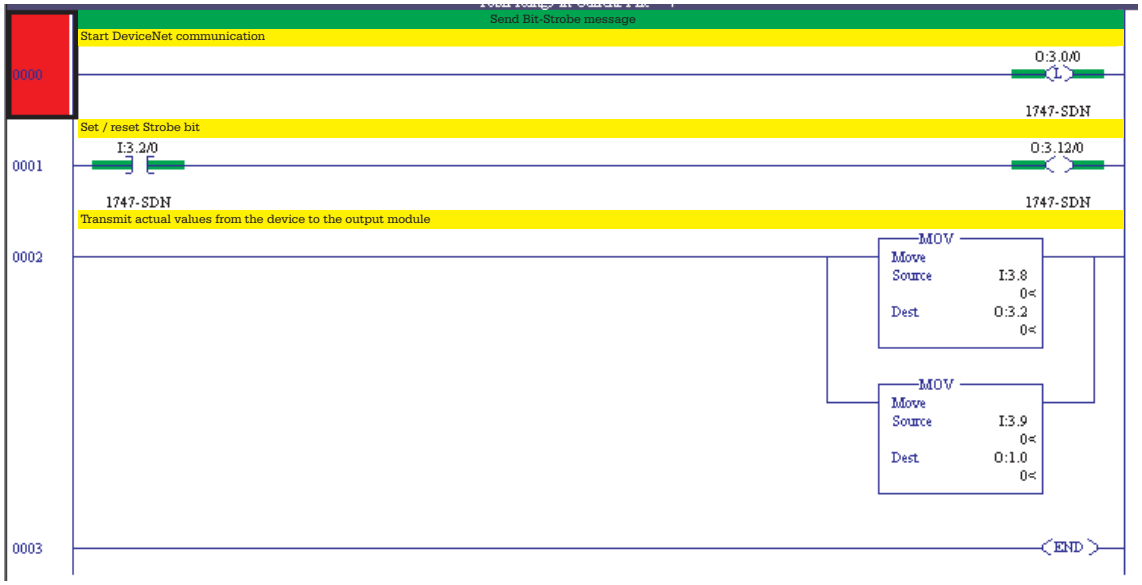
In the following program, the process input data are to be requested by MOVIDRIVE®. In addition, a fieldbus timeout should be triggered with the strobe-bit.

To do this, the following parameters must be set in the inverter:

Menu no.	Index	Parameters	Value
100	8461	Setpoint source	Fieldbus
101	8462	Control signal source	Fieldbus
870	8304	Process output data description 1	Control word 1
871	8305	Process output data description 2	Speed
872	8306	Process output data description 3	No function
873	8307	Process input data description 1	Status word 1
874	8308	Process input data description 2	Speed
875	8309	Process input data description 3	No function
876	8622	PO data enable	YES
831	8610	Fieldbus timeout response	Rapid stop/warn.

Table 20: Setting the parameters for process data exchange

The MOVIDRIVE® drive inverter is now in fieldbus mode and can receive process data.
The program can now be written for the SLC500.



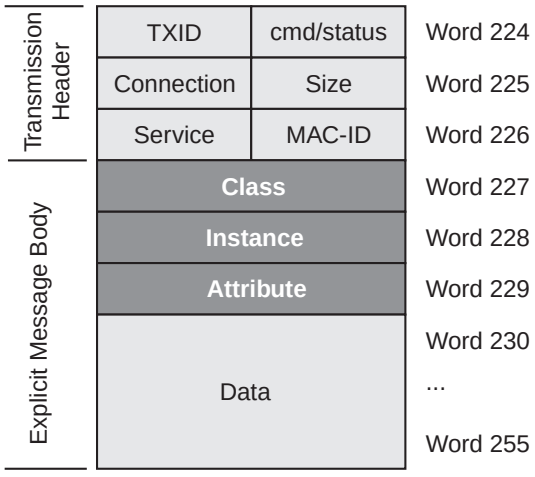
01915AEN

In rung 0, DeviceNet communication is once again enabled.
In rung 1, the strobe-bit of device 8 is set in relationship to the input bit I:3.2/0 (from the DeviceNet input module). Setting this bit triggers a fieldbus timeout in the inverter and the inverter changes to the rapid stop drive status.
In rung 2, process input data word 1 (I:3.8) is copied to the DeviceNet output module via memory word O:3.12, and process input data word 2 (I:3.9) is copied to the SLC500 output module via memory word O:1.0.

4.3 Exchange of explicit messages (parameter data)

Objective:

In this program, parameter data are to be exchanged between the control and the inverter.



01916AEN

Fig. 12: M-file

The exchange of parameter data between the inverter and the SLC500 is performed using files referred to as M-files (see the installation instructions for the DeviceNet scanner module).
A memory area from word 224 to 255 in these M-files is reserved for explicit messages. Fig. 12 shows the structure of this memory area.

This memory area is divided into two areas:
- Transmission header (3 words)
- Explicit message body

These memory areas are described in more detail in Table 21.

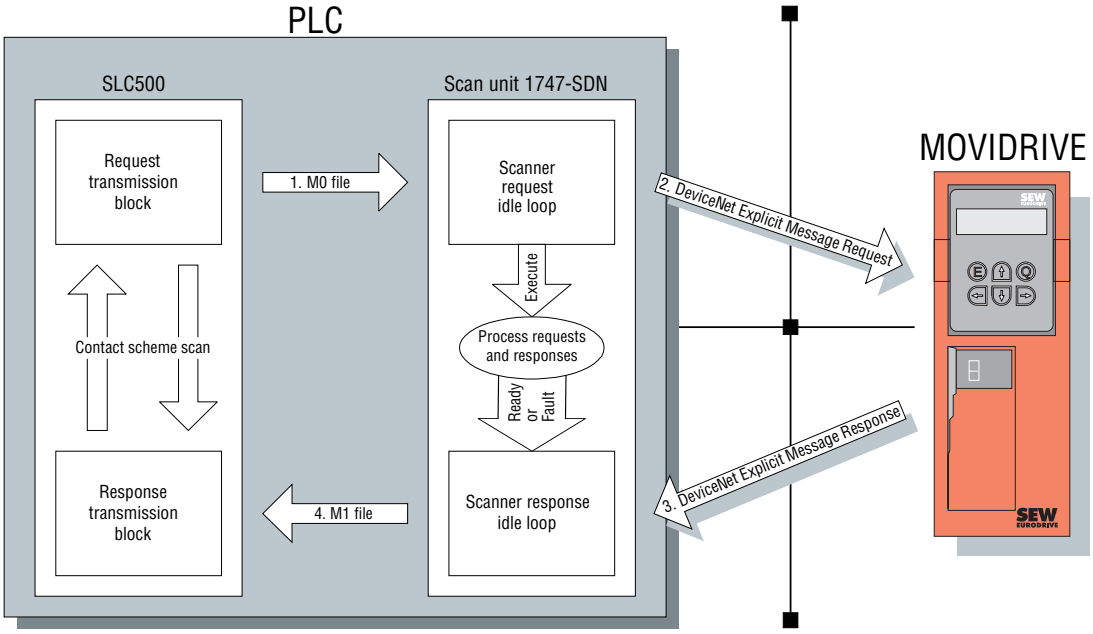
Memory area	Function	Length	Value	Description
Transmission header	cmd/status	½ word	See Table 22	cmd: Entry of a command code status: Entry of the transmission status
	TXID	½ word	1..255	During creation or downloading of a request to the scanner, the contact plan program of the SLC5 processor assigns a TXID to the transmission.
	Size	½ word	3..29	Size of the explicit message body (in bytes!!)
	Connection	½ word	0	DeviceNet connection (= 0)
	Service	½ word	0Eh 10h 05h etc.	Get_Attribut_Single (Read) Set_Attribut_Single (Write) Reset See DeviceNet specification for more services
Explicit message body	Class	1 word	0..255	DeviceNet class
	Instance	1 word	0..255	DeviceNet instance
	Attribute	1 word	0..255	DeviceNet attribute
	Data	0 – 26 words	0..65535	Data content

Table 21: Coding of the memory area within the M-file

Command code (cmd)	Description
0	Ignore transmssion block
1	Execute transmission block
2	Receive transmission status
3	Reset all client/server transmissions
4	Delete transmission from idle loop
5..255	Reserved
Network node status (status)	**Description**
0	Ignore transmission block
1	Transmission completed successfully
2	Transmission in progress
3	Error – Slave device not in the scan list
4	Error – Slave is off-line
5	Error – DeviceNet network connection deactivated (off-line)
6	Error – Unknown transmission TXID
7	Not used
8	Error – Invalid command code
9	Error – Scanner buffer full
10	Error – Other client/server transmission in progress
11	Error – No connection to slave device
12	Error – Response data are too long for the block
13	Error – Invalid connection
14	Error – Invalid size specified
15	Error – Occupied
16..255	Reserved

Table 22: Command and status codes

The M-files are divided into a request file (M0-file) and a response file (M1-file). The data transmission is shown in Fig. 13:



02148AEN

Fig. 13: Transmission of an explicit message

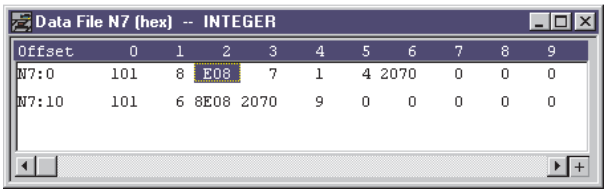
Über-tragungs-kopf	TXID	cmd/status	Wort 224
	Anschluss	Größe	Wort 225
	Dienst	MAC-ID	Wort 226
Explicit Message Body	**Class**		Wort 227
	Instance		Wort 228
	Attribute		Wort 229
	Index		Wort 230
	Datenwort Low (HEX)		Wort 231
	Datenwort High (HEX)		Wort 232

01919AEN

Fig. 14: SEW parameter channel

Register object class (7h) must be used in order to read (instance 1 to 8) or write (instance 2 and 3) parameters from the inverter via the SEW parameter data channel. The data range is divided into the index (2 bytes) and the parameter data (4 bytes).

In the sample program, a data area is reserved in the integer file (N-file), into which the data of the M0/M1 files are written.



02149AXX

Fig. 15: N-file

The data message which is to be sent is in N7:0 to N7:8, and N7:10 to N7:15 contain the data which have been received.

Request	
Function	**Value**
TXID	1
cmd	1 = Start
Connection	0
Size	8
Service	Eh = Read request
MAC_ID	8
Class	7
Instance	1
Attribute	4
Data 1	2070h
Data 2	0h
Data 3	0

Table 23a

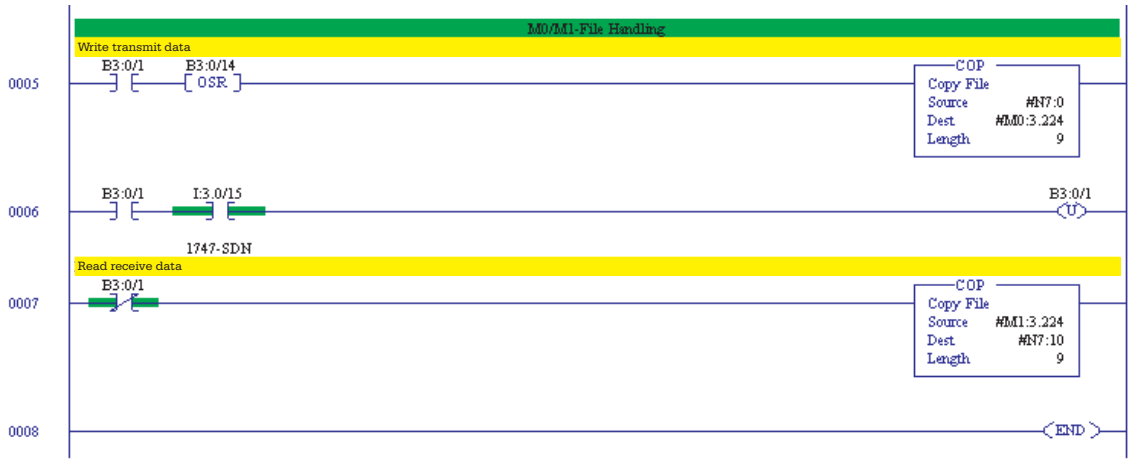
Response	
Function	**Value**
TXID	1
Status	1 = Successful
Connection	0
Size	6
Service	8Eh = Read response
MACID	8
Data 1	2070h
Data 2	9h
Data 3	0

Table 23b

The SEW parameter data channel can be addressed via class 7, instances 1 – 8 and attribute 4 (see statement of conformance).

In rung 5, the 9 bytes starting from N7:0 are copied into the M0-file with a rising edge of bit B3:0/1. This process starts the reading of parameter 8304 (2070h), followed by the program waiting for the rising edge of the scanner status bit I:3.0/15 in rung 6. This bit indicates that the data are present and request bit B3:0/1 can then be reset.

The received data still have to be written into the N-file. To this end, 9 words of the M1-file N7:10 – 19 are written.



01921AEN

5 Technical Data

Part no. 822 903 1

Number of process data words

- 1, 2 or 3 process data words
  can be selected using parameter P877 DeviceNet PD Configuration

Baud rate

- 125, 250 or 500 kbaud
  can be selected via DIP switch

Transmission level

- ISO 11 98 - 24 V

MAC-ID

- 0 ... 63
  can be selected via DIP switch
  Number of station: max. 64

Supported services

- Polled I/O: 1 ... 3 words
- Bit Strobe I/O: 1 ... 3 words
- Explicit messages: Get_Attribute_Single / Set_Attribute-Single / Reset / Allocate_M/S_Connection_Set / Release_M/S_Connection_Set

Supported objects

- Identity object class
- Message router class
- DeviceNet object class
- Connection object class
- Register object class
- Parameter object class

Communication protocol

- Master/slave connection set acc. to DeviceNet specification version 2.0

Connection system

- 2-wire bus and 2-wire supply voltage 24 VDC with 5-pole Phoenix terminal
- Pin assignment acc. to DeviceNet specification

Bus lengths

- 500 m at 125 kbaud
  250 m at 250 kbaud
  100 m at 500 kbaud
  for thick cable according to DeviceNet specification 2.0 Appendix B

6 Appendix

6.1 General error codes

General error code (hex)	Error name	Description
00 - 01		Reserved for DeviceNet
02	Resource unavailable	The source required for performing the service is unavailable
03 - 07		Reserved for DeviceNet
08	Service not supported	The service is not supported for the selected class/instance
09	Invalid attribute value	Invalid attribute data have been sent
0A		Reserved for DeviceNet
0B	Already in requested mode/state	The selected object is already in the requested mode/state
0C	Object state conflict	The selected object cannot perform the service in its current status
0D		Reserved for DeviceNet
0E	Attribute not settable	It is not possible to access the selected object for writing
0F	Privilege violation	Violation of access entitlement
10	Device state conflict	The current status of the device makes it impossible to perform the required service
11	Reply data too large	The length of the transferred data is longer than the size of the receiving buffer
12		Reserved for DeviceNet
13	Not enough data	The length of the transferred data is too short for the service to be performed
14	Attribute not supported	The selected attribute is not supported
15	Too much data	The length of the transferred data is too long for the service to be performed
16	Object does not exist	The selected object is not implemented in the device
17		Reserved for DeviceNet
18	No stored attribute data	The requested data have not been stored previously
19	Store operation failure	The data could not be stored because an error occurred whilst saving them
1A - 1E		Reserved for DeviceNet
1F	Vendor specific error	Proprietary error See the fieldbus device profile manual
20	Invalid parameter	Invalid parameter This error message is used when a parameter does not satisfy the requirements of the specification and/or the requirements of the application
21 - CF	Future extensions	Reserved by DeviceNet for additional definitions
D0 - FF	Reserved for Object Class and service errors	This area is intended for use if the error which has occurred cannot be assigned to any of the aforementioned error groups

Table 24: General error codes

6.2 Statement of conformance

DeviceNet Statement of Conformance

SOC data as of 9 - 16 - 1998

		Fill in the blank or [X] the appropriate box				
General Device Data	Conforms to DeviceNet Specification	Volume I - Release *2*		Volume II - Release *2*		
	Vendor Name	*SEW Eurodrive GmbH*				
	Device Profile Name	*Vendor Specific*				
	Product Name	*SEW-MOVIDRIVE-DFD11A*				
	Product Catalog Number	*3*				
	Product Revision	*1.01*				
DeviceNet Physical Conformance Data	Network Power Consumption (Max)	*1 A @ 11V dc (worst case)*				
	Connector Style	Open-Hardwired	[ ]	Sealed-Mini	[ ]	
		Open-Pluggable	[X]	Sealed-Micro	[ ]	
	Isolated Physical Layer	Yes	[X]			
		No	[ ]			
	LEDs Supported	Module	[ ]	Combo Mod/Net	[X]	
	None [ ]	Network	[ ]	I/O	[ ]	
	MAC ID Setting	DIP Switch	[X]	Software Settable	[ ]	
		Other				
	Default MAC ID	*63*				
	Communication Rate Setting	DIP Switch	[X]	Software Settable	[ ]	
		Other				
	Communication Rates Supported	125k bit/s	[X]	500k bit/s	[X]	
		250k bit/s	[X]			
DeviceNet Communication Data	Device Network Behavior	Group 2 Client	[ ]	Group 2 Only Client	[ ]	
	Check All That Apply	Group 2 Server	[ ]	Group 2 Only Server	[X]	
		Peer-To-Peer	[ ]	Tool (not a Device)	[ ]	
	UCMM Explicit Message Groups Supported	Group 1	[ ]	Group 2 [ ]	Group 3	[ ]
	Dynamic I/O Message Groups (Peer to Peer)	Group 1	[ ]	Group 2 [ ]	Group 3	[ ]
	Default I/O Data Address Path	Input: Class *4*		Inst. *64*	Attr. *3*	
		Output: Class *4*		Inst. *64*	Attr. *3*	
	Fragmented Explicit Messaging Supported	Yes	[X]		No	[ ]
	If yes, Acknowledge TimeOut	*1000 ms*				
	Typical Target Addresses					
	Consumption Service *16*	Class *1*		Inst. *1*	Attr. *7*	
	Production Service *14*	Class *1*		Inst. *1*	Attr. *7*	

1 of *9*

DeviceNet Statement of Conformance

DeviceNet Required Object Implementation	Identity Object 0x01					
	Object Class	ID	Description	Get	Set	Value Limits
	Attributes Open	1	Revision	☐	☐	
		2	Max instance	☐	☐	
	[X] None Supported	3	Number of Instances	☐	☐	
		4	Optional attributes list	☐	☐	
		5	Optional services list	☐	☐	
		6	Max Id of class attributes	☐	☐	
		7	Max Id of instance attributes	☐	☐	
		DeviceNet Services		Parameter Options		
	Services	☐	Get_Attributes_All			
		☐	Reset			
	[X] None Supported	☐	Get_Attribute_Single			
		☐	Find_Next_Object_instance			
	Object Instance	ID	Description	Get	Set	Value Limits
	Attributes Open	1	Vendor	[X]	☐	=(315)
		2	Device type	[X]	☐	=(100)
		3	Product code	[X]	☐	=(3)
		4	Revision	[X]	☐	=(1.01)
		5	Status (bits supported)	[X]	☐	
		6	Serial number	[X]	☐	=(8229031)
		7	Product name	[X]	☐	SEW-MOVIDRIVE-DFD11A
		8	State	☐	☐	
		9	Config. Consistency Value	☐	☐	
		10	Heartbeat Interval	☐	☐	
		DeviceNet Services		Parameter Options		
	Services	☐	Get_Attributes_All			
		[X]	Reset	0		
		[X]	Get_Attribute_Single			
		☐	Set_Attribute_Single			
	Vendor Specific Additions	If yes, fill out the Vendor Specific Additions form on page F-7.		Yes	☐	
				No	[X]	

[X] Get to indicate that attribute value is returned by the use of Get_Attribute_Single service.

[X] Set to indicate that attribute value is written to by the use of Set_Attribute_Single service.

2 of 9

DeviceNet Statement of Conformance

DeviceNet Required Object Implementation	Message Router Object 0x02					
	Object Class		ID	Description	Get	Set	Value Limits
	Attributes	Open	1	Revision	☐	☐	
			4	Optional attribute list	☐	☐	
	☒ None Supported		5	Optional service list	☐	☐	
			6	Max ID of class attributes	☐	☐	
			7	Max ID of instance attributes	☐	☐	
			DeviceNet Services		Parameter Options		
	Services		☐	Get_Attributes_All			
			☐	Get_Attribute_Single			
	☒ None Supported						
	Object Instance		ID	Description	Get	Set	Value Limits
	Attributes	Open	1	Object list	☐	☐	
			2	Maximum connections supported	☐	☐	
	☒ None Supported		3	Number of active connections	☐	☐	
			4	Active connections list	☐	☐	
			DeviceNet Services		Parameter Options		
	Services		☐	Get_Attributes_All			
			☐	Get_Attribute_Single			
	☒ None Supported						
	Vendor Specific Additions		If yes, fill out the Vendor Specific Additions form on page F-7.		Yes	☐	
					No	☒	

☒ **Get** to indicate that attribute value is returned by the use of Get_Attribute_Single service.

☒ **Set** to indicate that attribute value is written to by the use of Set_Attribute_Single service.

DeviceNet Statement of Conformance

DeviceNet Required Object Implementation	DeviceNet Object 0x03					
	Object Class	ID	Description	Get	Set	Value Limits
	Attributes Open	1	Revision	X	☐	=(2)
	☐ None Supported					
		DeviceNet Services		Parameter Options		
	Services	☐	Get_Attribute_Single			
	X None Supported					
	Object Instance	ID	Description	Get	Set	Value Limits
	Attributes Open	1	MAC ID	X	☐	=(0..63)
		2	Baud rate	X	☐	=(0..2)
	☐ None Supported	3	BOI	X	☐	=(0)
		4	Bus-off counter	X	X	=(0..255)
		5	Allocation information	X	☐	
		6	MAC ID switch changed	X	☐	=(0)
		7	Baud rate switch changed	X	☐	=(0)
		8	MAC ID switch value	X	☐	=(0..63)
		9	Baud rate switch value	X	☐	=(0..2)
		DeviceNet Services		Parameter Options		
	Services	X	Get_Attribute_Single			
		X	Set_Attribute_Single			
	☐ None Supported	X	Allocate M/S connection set			
		X	Release M/S connection set			
	Vendor Specific Additions	If yes, fill out the Vendor Specific Additions form on page F-7.		Yes	☐	
				No	X	

X **Get** to indicate that attribute value is returned by the use of Get_Attribute_Single service.

X **Set** to indicate that attribute value is written to by the use of Set_Attribute_Single service.

4 of 9

DeviceNet Statement of Conformance

DeviceNet Required Object Implementation

Connection Object 0x05

Object Class

Attributes: Open
[X] None Supported

ID	Description	Get	Set	Value Limits
1	Revision	[ ]	[ ]	

DeviceNet Services — **Parameter Options**

Services
[X] None Supported

- [ ] Reset
- [ ] Create
- [ ] Delete
- [ ] Get_Attribute_Single
- [ ] Find_Next_Object_Instance

Object Instance

Complete the Object Instance section for each Instance type supported. Indicate Production trigger, Transport type and Transport Class supported for Dynamic I/O.

Predefined M/S Connections

- Explicit Message [X]
- Polled [ ]
- Bit Strobed [ ]
- Change of State [ ]
- Cyclic [ ]

Peer to Peer Connections

- Explicit Message [ ] — Server: Total / Client
- Dynamic I/O [ ] — Server: Total / Client

Max Instances

Production trigger(s)	Cyclic	[ ]	COS	[ ]	App. trig.	[ ]
Transport type(s)	Server	[X]			Client	[ ]
Transport class(es)		[ ]	2	[X]	3	[ ]

Attributes: Open

ID	Description	Get	Set	Value Limits
1	State	[X]	[ ]	
2	Instance type	[X]	[ ]	*0*
3	Transport Class trigger	[X]	[ ]	*131*
4	Produced connection ID	[X]	[ ]	
5	Consumed connection ID	[X]	[ ]	
6	Initial comm. characteristics	[X]	[ ]	*33*
7	Produced connection size	[X]	[ ]	*16*
8	Consumed connection size	[X]	[ ]	*16*
9	Expected packet rate	[X]	[X]	*=(0..65530)*
12	Watchdog time-out action	[X]	[ ]	*1*
13	Produced connection path length	[X]	[ ]	*0*
14	Produced connection path	[X]	[ ]	
15	Consumed connection path length	[X]	[ ]	*0*
16	Consumed connection path	[X]	[ ]	
17	Production inhibit time	[X]	[ ]	*=(0)*

DeviceNet Services — **Parameter Options**

Services

- [X] Reset
- [ ] Delete
- [ ] Apply_Attributes
- [X] Get_Attribute_Single
- [X] Set_Attribute_Single

Vendor Specific Additions

If yes, fill out the Vendor Specific Additions form on page F-7.

Yes [ ]
No [X]

[X] **Get** to indicate that attribute value is returned by the use of Get_Attribute_Single service.
[X] **Set** to indicate that attribute value is written to by the use of Set_Attribute_Single service.

5 of *9*

DeviceNet Statement of Conformance

DeviceNet Required Object Implementation

Connection Object 0x05

Object Class	ID	Description	Get	Set	Value Limits
Attributes Open	1	Revision	☐	☐	
☒ None Supported					

	DeviceNet Services	Parameter Options
Services	☐ Reset	
	☐ Create	
☒ None Supported	☐ Delete	
	☐ Get_Attribute_Single	
	☐ Find_Next_Object_Instance	

Object Instance

Complete the Object Instance section for each Instance type supported. Indicate Production trigger, Transport type and Transport Class supported for Dynamic I/O.

Predefined M/S Connections		Peer to Peer Connections		Max Instances
Explicit Message	☐	Explicit Message	☐	Total
Polled	☒	Server		Client
Bit Strobed	☐	Dynamic I/O	☐	Total
Change of State	☐	Server		Client
Cyclic	☐			

Production trigger(s)	Cyclic	☐	COS	☐	App. trig.	☐
Transport type(s)	Server	☒			Client	☐
Transport class(es)		☐	2	☒	3	☐

Attributes Open	ID	Description	Get	Set	Value Limits
	1	State	☒	☐	
	2	Instance type	☒	☐	*1*
	3	Transport Class trigger	☒	☐	*130*
	4	Produced connection ID	☒	☐	
	5	Consumed connection ID	☒	☐	
	6	Initial comm. characteristics	☒	☐	*1*
	7	Produced connection size	☒	☒	*2,4,6*
	8	Consumed connection size	☒	☒	*2,4,6*
	9	Expected packet rate	☒	☒	*=(0..65530)*
	12	Watchdog time-out action	☒	☐	*0*
	13	Produced connection path length	☒	☐	*6*
	14	Produced connection path	☒	☐	
	15	Consumed connection path length	☒	☐	*6*
	16	Consumed connection path	☒	☐	
	17	Production inhibit time	☒	☐	*=(0)*

	DeviceNet Services	Parameter Options
Services	☒ Reset	
	☐ Delete	
	☐ Apply_Attributes	
	☒ Get_Attribute_Single	
	☒ Set_Attribute_Single	

Vendor Specific Additions			
	If yes, fill out the Vendor Specific Additions form on page F-7.	Yes	☐
		No	☒

☒ **Get** to indicate that attribute value is returned by the use of Get_Attribute_Single service.

☒ **Set** to indicate that attribute value is written to by the use of Set_Attribute_Single service.

6 of *9*

DeviceNet Statement of Conformance

DeviceNet Required Object Implementation

Connection Object 0x05

Object Class

Attributes: Open
[X] None Supported

ID	Description	Get	Set	Value Limits
1	Revision	[ ]	[ ]	

DeviceNet Services — **Parameter Options**

Services
[X] None Supported

- [ ] Reset
- [ ] Create
- [ ] Delete
- [ ] Get_Attribute_Single
- [ ] Find_Next_Object_Instance

Object Instance

Complete the Object Instance section for each Instance type supported. Indicate Production trigger, Transport type and Transport Class supported for Dynamic I/O.

Predefined M/S Connections

- Explicit Message [ ]
- Polled [ ]
- Bit Strobed [X]
- Change of State [ ]
- Cyclic [ ]

Peer to Peer Connections

- Explicit Message [ ] — Server: Total / Client
- Dynamic I/O [ ] — Server: Total / Client

Max Instances

Production trigger(s)	Cyclic	[ ]	COS	[ ]	App. trig.	[ ]
Transport type(s)	Server	[X]			Client	[ ]
Transport class(es)		[ ]	2	[X]	3	[ ]

Attributes: Open

ID	Description	Get	Set	Value Limits
1	State	[X]	[ ]	
2	Instance type	[X]	[ ]	
3	Transport Class trigger	[X]	[ ]	
4	Produced connection ID	[X]	[ ]	
5	Consumed connection ID	[X]	[ ]	
6	Initial comm. characteristics	[X]	[ ]	*2*
7	Produced connection size	[X]	[X]	*2,4,6*
8	Consumed connection size	[X]	[ ]	
9	Expected packet rate	[X]	[X]	*=(0..65530)*
12	Watchdog time-out action	[X]	[ ]	*0*
13	Produced connection path length	[X]	[ ]	
14	Produced connection path	[X]	[ ]	
15	Consumed connection path length	[X]	[ ]	
16	Consumed connection path	[X]	[ ]	
17	Production inhibit time	[X]	[ ]	*=(0)*

DeviceNet Services — **Parameter Options**

Services

- [X] Reset
- [ ] Delete
- [ ] Apply_Attributes
- [X] Get_Attribute_Single
- [X] Set_Attribute_Single

Vendor Specific Additions

If yes, fill out the Vendor Specific Additions form on page F-7.

Yes [ ]
No [X]

[X] **Get** to indicate that attribute value is returned by the use of Get_Attribute_Single service.
[X] **Set** to indicate that attribute value is written to by the use of Set_Attribute_Single service.

7 of *9*

DeviceNet Statement of Conformance

DeviceNet Required Object Implementation	Register Object 0x07					
	Object Class	ID	Description	Get	Set	Value Limits
	Attributes Open	1	Revision	☐	☐	
	☒ None Supported					
		DeviceNet Services		Parameter Options		
	Services	☐	Get_Attribute_Single			
	☒ None Supported					
	Object Instance	ID	Description	Get	Set	Value Limits
	Attributes Open	1	Bad Flag	☒	☐	
		2	Direction	☒	☐	
	☐ None Supported	3	Size	☒	☐	=(16,48)
		4	Data	☒	☒	>(65536)
		DeviceNet Services		Parameter Options		
	Services	☒	Get_Attribute_Single	8300		
	☐ None Supported	☒	Set_Attribute_Single			
	Vendor Specific Additions	If yes, fill out the Vendor Specific Additions form on page F-7.		Yes	☐	
				No	☒	

☒ **Get** to indicate that attribute value is returned by the use of Get_Attribute_Single service.

☒ **Set** to indicate that attribute value is written to by the use of Set_Attribute_Single service.

8 of 9

DeviceNet Statement of Conformance

Parameter Object 0x0F

DeviceNet Required Object Implementation		ID	Description	Get	Set	Value
Object Class						
Attributes	Open	1	Revision	☐	☐	
		2	Max instance	☒	☐	
☐ None Supported		8	Parameter class descriptor	☒	☐	
		9	Configuration assembly instance	☒	☐	
		10	Native language	☐	☐	
		DeviceNet Services		Parameter Options		
Services		☐	Get_Attributes_All			
		☐	Reset			
☐ None Supported		☒	Get_Attribute_Single			
		☐	Set_Attribute_Single			
		☐	Restore			
		☐	Save			
Object Instance		ID	Description	Get	Set	Value
Attributes	Open	1	Parameter value	☒	☒	*=(0..42*
		2	Link Path size	☒	☐	*0*
☐ None Supported		3	Link path	☒	☐	
		4	Descriptor	☒	☐	*=(16,4*
		5	Data type	☒	☐	*25*
		6	Data size	☒	☐	*=(4)*
		7	Parameter name string	☐	☐	
		8	Units string	☐	☐	
		9	Help string	☐	☐	
		10	Minimum value	☐	☐	
		11	Maximum value	☐	☐	
		12	Default value	☐	☐	
		13	Scaling multiplier	☐	☐	
		14	Scaling divisor	☐	☐	
		15	Scaling base	☐	☐	
		16	Scaling offset	☐	☐	
		17	Multiplier link	☐	☐	
		18	Divisor link	☐	☐	
		19	Base link	☐	☐	
		20	Offset link	☐	☐	
		21	Decimal precision	☐	☐	
		DeviceNet Services		Parameter Options		
Services		☐	Get_Attribute_All			
☐ None Supported		☒	Get_Attribute_Single			

6.3 Definitions of terminology

Term	Description
Allocate	Provides a service for setting up a connection
Attribute	Attribute of an object class or instance. Describes the characteristics of the object class or instance more fully.
BIO - Bit-strobe I/O	All participants can be addressed with a broadcast message. The addressed participants respond with the process input data.
Class	DeviceNet object class
DeviceNet scanner	Plug-in module for the Allen Bradley PLC which connects the PLC fieldbus to the peripheral devices
DUP-MAC check	Duplicate MAC-ID test
Explicit message body	Includes the class no., instance no., attribute no. and the data
Explicit message	Parameter data telegram; assists in addressing the DeviceNet objects
Get_Attribute_Single	Read service for a parameter
Instance	Instance of an object class. Divides the object classes into additional subgroups.
MAC-ID	Media Access Control identifier: Node address of the device
M-file	Provides the data range between the PLC and the scanner module
Mod/Net	Module/network
Node ID	Node address = MAC-ID
PIO - Polled I/O	Process data channel of DeviceNet; allows process output data to be sent and process input data to be received
Release	Provides a service for dropping a connection
Reset	Provides a service for resetting an error
Rung	SLC500 program line
Service	Service performed over the bus, e.g. read service, write service, etc.
Set_Attribute_Single	Write service for a parameter
SLC500	Allen Bradley PLC

Table 25: Definitions of terminology

7 Index

B
Bit-strobe I/O 18
Bus termination 7

D
Definitions of terminology 46
DeviceNet network, setting up 13
DIP switches
Setting 8
Display elements 8
BIO LED 10
BUSOFF LED 10
ModNet LED 9
PIO LED 9
Power-up 8

E
EDS file 13
Equipotential bonding 7
Example 26
Bit-strobe I/O 30
Explicit messages (parametric data) 31
Polled I/O (process data) 28

G
General error codes 36

I
Installation 6
Bus cables 7
Inverter control mode fieldbus 11

P
Pin assignment 6
Polled I/O 14
Process data exchange 14
Project planning 11
Project planning for x process data words 15

S
Startup 11
Statement of conformance 37
Supported unit types 6

T
Timeout of explicit messages 25
Timeout response with bit-strobe I/O 19
Timeout response with polled I/O 17

We are available, wherever you need us.
Worldwide.

SEW-EURODRIVE right around the globe is your competent partner in matters of power transmission with manufacturing and assembly plants in most major industrial countries.

SEW
EURODRIVE

SEW-EURODRIVE GmbH & Co · P.O.Box 30 23 · D-76642 Bruchsal/Germany
Tel. +49-7251-75-0 · Fax +49-7251-75-19 70 · Telex 7 822 391
http://www.SEW-EURODRIVE.com · sew@sew-eurodrive.com